ARMY RESEARCH LABORATORY

Model Evaluation Tool

by Stephen F. Kirby

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

Citation of manufacturers' or trade names does not constitute an official endorsement or approval of the use thereof.

Army Research Laboratory

White Sands Missile Range, NM 88002-5501

Model Evaluation Tool

Stephen F. Kirby
Computational and Information Sciences Directorate
Battlefield Environment Division

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
November 2004	Final	Oct 2003 – Sep 2004

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Model Evaluation Tool	
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
Stephen F. Kirby	
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
U.S. Army Research Laboratory Computational and Information Sciences Directorate Battlefield Environment Division (ATTN: AMSRD-ARL-CI-EB) White Sands Missile Range, NM 88002-5501	ARL-TR-3373

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
U.S. Army Research Laboratory 2800 Powder Mill Road Adelphi, MD 20783-1145	
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)
	ARL-TR-3373

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

A Web-based Model Evaluation Tool (MET) has been developed that allows a user to 1) gather initialization data, and 2) run and statistically analyze output from these mesoscale models: Mesoscale Model Version 5 (MM5) and the Weather Research and Forecasting model. The models may be run as single entities or as an ensemble. An example of an ensemble would be a multi-analysis ensemble: running MM5 using the Aviation, Eta, Medium-Range Forecasting, and Naval Operational Global Atmospheric Prediction System models for large-scale initialization. If an ensemble is run, weights may be assigned to each ensemble member parameter based on the Newbold-Granger scheme. In addition, the user is able to assimilate other data types into the model runs, such as raobs and surface data. Also discussed is an initial effort to add a mobile capability for MET, allowing access through a cell phone.

15. SUBJECT TERMS

Web Application, Java Server Pages, Java Server Faces, Java Web Services, Mesoscale Model Ensemble, Statistical Analysis

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UU	32	Stephen F. Kirby
U	U	U			**19b. TELEPHONE NUMBER** (Include area code) (505) 678-7474

Contents

List of Figures

Acknowledgments

INTENTIONALLY LEFT BLANK.

Summary

A Web-based software system, the Model Evaluation Tool (MET), is being developed to both simplify the process of mesoscale weather model evaluation and empower researchers by placing an array of model run options, data sources, and display options at their fingertips.

Through the Web interface, a user selects such entities as the model run mode (i.e., single model or ensemble) and whether or not to employ data assimilation. Once all selections are complete, a large-scale initialization data download begins automatically. Upon completion of this task, the model runs begin, followed by an optional display. MET is currently hosted on a 32-node Linux cluster.

The two mesoscale models that may be analyzed using MET are the Mesoscale Model Version 5 (MM5) and the Weather Research and Forecasting (WRF) Model. MET provides two ways to run using an ensemble method. The first ensemble mode is the "multi-analysis" method, whereby a mesoscale model is initialized with a number of different large-scale initialization datasets. The other ensemble mode is the multi-model method, whereby output from two or more different mesoscale models is combined in some fashion. Other data types may be assimilated at run time, including radiosonde observations (raobs) and surface meteorological data in the form of Meteorological Aerodrome Reports (METAR). Regardless of the data type chosen for assimilation, an intermediary program is run to determine the quality of both raob and METAR data prior to assimilation. Data flagged as bad are not used.

Upon model run completion, various statistical analyses of the output, dependent on the model run mode selected, are carried out. For the individual ensemble members, numerous other statistical values are computed including bias, root mean square error, dispersion, and rank correlation.

From the MET Web interface, using a composition of Web services, a user can use the MET to determine an optimum model paradigm for a selected parameter or parameter group, by running the MM5 and WRF in the single model and in the two ensemble modes. Each Web service has the task of determining the bias values for a given parameter for a particular model mode. Currently, MET can only verify model output in the western United States using the MesoWest mesonet dataset; other test regions will be added in future enhancements to MET.

A "lightweight" capability, whereby MET can be accessed via cell phone, is also being constructed, using Wireless Markup Language cards that facilitate display on the small screen of a typical cell phone. Connectivity to the server is done through Wireless Application Protocol.

1. Overview

1.1   Introduction

A Model Evaluation Tool (MET) has been developed that allows a user to perform the following:

1. Gather large-scale initialization datasets, including these models: Aviation (AVN), Eta, Medium Range Forecasting (MRF), and Naval Operational Global Atmospheric Prediction System (NOGAPS).

2. Assimilate data sources, such as raobs and surface observations, into model runs.

3. Execute MM5 or WRF model runs in parallel on a Linux cluster.

4. Generate statistics, both at the surface and at vertical levels, including model bias, rank correlation, root mean square error, etc., for temperature, *u*- and *v*-components of the wind, and relative humidity. Also, when run as an ensemble, be able to generate Newbold-Granger weighting coefficients for each ensemble member parameter.

5. Display statistics employing the Visualization for Algorithm Development (VisAD) toolkit (*1*).

Java Server Pages (JSP) form the foundation of this tool by orchestrating the program flow and much of the visual display. Java Server Faces (JSF) has recently been instituted specifically for the user interface.

Statistical results can be converted into Extensible Markup Language (XML) and stored in a freeware XML database, called eXist. The Java Application Programming Interface (API) for XML Processing (JAXP) is used for the task of converting XML for display as Hypertext Markup Language (HTML) with the application of stylesheets. (*2*)

In addition, initial work has begun on making MET available through a mobile device, such as a cell phone, through the use of a freeware gateway server and Wireless Markup Language (WML).

1.2   Data Assimilation

Within MET, data assimilation is possible with either the MM5 or the WRF, and the model is also able to assimilate raobs as well as surface reports. The MM5 has adopted a text format to be utilized in data assimilation, called LITTLE_R, which can be a concatenation of surface reports, raob data, satellite data, etc. For example, when a MET user acquires raob data, it is first passed to a C program that reads the raob text data, generates FORTRAN code, and then compiles and runs the code, thus creating a LITTLE_R format file of concatenated raobs. At this point, a built-in routine, called 3DVAR_OSPROC, is run. Its purpose is to "filter" the LITTLE_R data file for duplicate levels, superadiabatic regions, etc. This file now serves as the observations (commonly referred to as the "ob") file for assimilation. Since the WRF is a descendant of the MM5, it too has adopted the LITTLE_R format for data assimilation input, although in the long term, the Binary Universal Form for the Representation of meteorological data will likely

supersede LITTLE_R in order to make WRF formats more standardized. Unlike the MM5, the WRF does not require a **first guess** field when carrying out the LITTLE_R data filtering function (i.e., a check for missing observations, superadiabatic lapse rates, etc.).

1.3 Program flow

Figure 1 is a flowchart depicting the chain of events within MET when a user requests an MM5 run. With the exception of the first code segment (top.html), all of the files are built on JSP, which can broadly be described as HTML with embedded Java code.
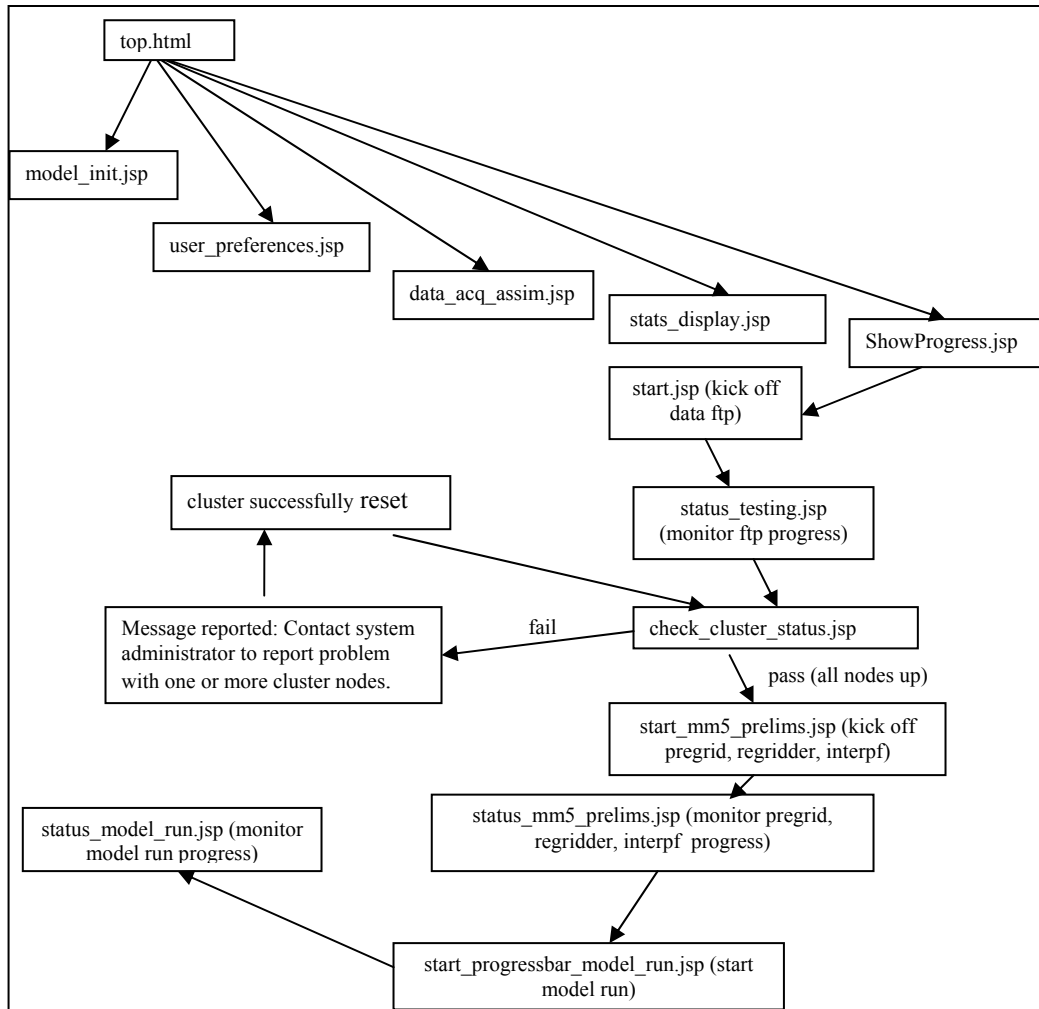
Figure 1. MET process flowchart.

The code segments perform the following functions:

1. top.html – This serves as the main page with links to three other JSP pages. This page renders a page with five user tabs: 1) Model Selection/Physics, 2) User Settings, 3) Data Acquisition/Assimilation, 4) Statistical Analysis/Display, and 5) Start Model Initialization Data FTP. When a user selects one of the tabs, it links to the following JSP, respectively: model_init.jsp, user_preferences.jsp, data_acq_assim.jsp, stats_display.jsp, and ShowProgress.jsp. Each of these JSP utilizes JSF to generate menus.

The JSP, model_init.jsp, presents the user with a tabbed pane of menus for the following topics: model, cumulus scheme, implicit moisture scheme, and planetary boundary layer. The **Model** menu pane yields a radio button list of options including "MM5 multi-analysis ensemble," "MM5 initialized with NOGAPS," "WRF initialized with AVN," and "WRF initialized with Eta." Similarly, the **Cumulus Scheme** pane allows the user to select from "None," "Kuo," "Grell," etc. The **Implicit Moisture Scheme** pane provides the options of "Dry," "Stable," "Warm," etc., and lastly, the **Planetary Boundary Layer** pane yields selections of "None," "Bulk," "Blackadar," etc. At any time during this process, the user can select the **Submit Values/Check Current Settings** button at the bottom of each selection window or may leave the tabbed pane selection area completely by selecting the **Home** button, which returns the user to the main page of user options. If the **Submit Values/Check Current Settings** button is selected, a table is rendered indicating the user's current selections. Selecting the User Settings tab from the main page yields a JSF page with three menus: **Model Run Location**, **Number of Forecast Hours**, and **Parameter to Optimize for**. Because the statistical analysis routines are currently set to use the MesoWest mesonet (*3*) data for "ground truth," only one location is currently available: Utah and the surrounding area. The user selects the number of forecast hours from a drop-down menu, and then, the parameter(s) to optimize for. The choices consist of temperature, relative humidity, the *u*-component of the wind, the *v*-component of the wind, winds overall (which considers both *u*- and *v*-components), and all parameters (which considers temperature, relative humidity, and winds). Determining the optimum parameter within MET is controlled by a series of Web services that work in concert, controlled by a Java servlet, and is discussed in the section 4.

Choosing the "Data Acquisition/Assimilation" tab from the main page provides two menus. The **Data Acquisition** menu allows the user to choose from raobs from University of Wyoming or surface observations (decoded Meteorological Aerodrome Reports (METAR) messages) from Florida State University. The **Data Assimilation** menu provides two options: none, or surface observations and raobs. The "Statistical Analysis/Display" tab points to a page with two menu panes: **Scatterplots** and **Bias Plots**. The **Scatterplots** menu can generate graphics for winds, temperature, or relative humidity, and display ground truth versus model values over a 24-h forecast period. The Bias Plots menu provides graphics for the same parameters and represents "model-truth" values over a 24-h forecast. Once the menu selections are complete, the information is passed to a JavaBean, which processes the information and sets the appropriate parameters. The ShowProgress.jsp calls the start.jsp to initialize the retrieval of model initialization data.

2. start.jsp – This function initializes the process of using FTP to acquire AVN, Eta, and MRF data by starting two Java threads: one to manage the overall process and one to execute the various file transfers.

3. status_testing.jsp – This function performs the FTP of AVN, Eta, and MRF datasets and tracks the progress of each acquisition by using JavaScript to refresh a progress bar drawn each second. When all the FTP sessions are complete, the check_cluster_status.jsp function is called.

4. check_cluster_status.jsp – Upon completion of the file transfers, the Scali (parallel processing software) scaconftool utility is called to determine the status of the 32 cluster

nodes. Upon successful return, the execution of three MM5 preprocessing routines—PREGRID, REGRIDDER, and INTERPF—begins.

5. start_mm5_prelims.jsp – This function initializes the process of executing PREGRID, REGRIDDER, and INTERPF by starting a Java thread and setting counters for the various processes to 0.

6. status_mm5_prelims.jsp – This function executes PREGRID, REGRIDDER, and INTERPF and also draws a progress bar to the screen that tracks file sizes and uses JavaScript to redraw the progress bar every second.

7. start_progressbar_model_run.jsp – As with previous start*.jsp modules, this function initializes the model run process by starting a Java thread.

8. status_model_run.jsp – This function kicks off a Java thread to start the model run and renders a progress bar to track model output file sizes, again using JavaScript to do so.

9. ensemble_stats.jsp – This function renders results pertaining to the model, which yields the optimum values for the parameter the user selected as determined by the composition of Web services.

Throughout this process, a file key to JSF—faces-config.xml—helps to orchestrate the program flow by acting as a gatekeeper. It allows program control to move from one HTML page to a JSP or from one JSP to another JSP, etc., but only if it receives the required value back from a JavaBean, permitting transition to occur. For example, program control in MET is passed from the login JSP to the greeting JSP only if the relevant JavaBean passes back the value SUCCESS to the faces-config.xml file.

2. Ensembles

Mesoscale models can be run standalone or as ensembles. Model ensembles were introduced with the theory that a single model run with one type of initialization dataset cannot possibly capture the true state of the atmosphere, and that by using multiple models or multiple initialization datasets, it is more likely that the true phase space of the atmosphere can be encompassed. Furthermore, an analysis cannot exactly match the initial state of the atmosphere and it is more than likely that model error is introduced by any one of a model's physical parameterizations. In an attempt to surmount these problems, model ensembles have been introduced.

An ensemble can be run in one of three modes:

- Multi-model: This mode allows the user to run two or more models jointly and combine the outputs.

- Multi-analysis: This mode allows the user to run one model, but vary the inputs for each run.

- Multi-model, multi-analysis: This mode allows the user to incorporate more than one model and vary the initialization data.

When ensembles are run, MET can also generate Newbold-Granger (N-G) (4) weighting coefficients as formulated in eq 1:

$$w_{jf} = [(\sum_{t=T-v}^{T-1} (X_{Tf} - X_{jf}))]^{-2} / \sum_{k} [(\sum_{t=T-v}^{T-1} X_{Tf} - X_{kf})]^{-2} \qquad (1)$$

Thus, the weight for a particular forecast is represented by the inverse square of the differences between the truth for the forecast and the input forecast value summed over a given time window, divided by the inverse square of the difference between the truth for the forecast and the ensemble member value summed over all ensemble members. In this way, these weights are normalized. Here, the forecast value for a given parameter is the model output value bilinearly interpolated to the location of the truth (mesonet) value, thus allowing for direct comparison. The greater the accuracy of a model forecast parameter over some time window (a 24-h window was used in initial testing), the higher the weight it will be assigned. These weights can then be carried into a subsequent forecast cycle with the assumption that the trend just calculated in terms of N-G weights will carry forward. A fused forecast (5) can be created as given in eq 2.

$$X_f = (\sum_{j=1}^{n} w_{jf} c_{jf} (X_{jf} + b_{jf})) / (\sum_{j=1}^{n} w_{jf} c_{jf}) \qquad (2)$$

where

f = variable

X_f = forecast

X_{jf} = input forecast

c_{jf} = associated confidence

w_{jf} = weights

b_{jf} = biases

For simplicity, the biases are assumed to be 0 and the confidences 1.

One statistic calculated in MET is ensemble spread, which gives an indication of how well the ensemble is performing (5). It is given by

$$SP(f) = \sqrt{1/N \sum_{n=1}^{N} (f_{i,j}^n - \tilde{f}_{i,j})^2}^{i,j} \qquad (3)$$

$$\overline{f_{i,j}}^{i,j} = 1/(I \bullet J) \sum_{i=1}^{I} \sum_{j=1}^{J} f_{i,j} \qquad (4)$$

$$\tilde{f}_{i,j} = 1/N \sum_{n=1}^{N} f_{i,j}^n \qquad (5)$$

with *f* being some forecast parameter and *N* the number of ensemble members. A small spread is generally, but not necessarily, indicative of a more accurate forecast.

3. General (Non-Ensemble) MET statistics

For both the MM5 and WRF, statistical values may be computed for surface parameters as well as for those at all of the pressure levels up to model top. Surface analysis model parameters are bilinearly interpolated to mesonet (only the MesoWest mesonet is currently available with MET) locations. In the case of the MM5, a FORTRAN program extracts only the forecast data output on the hour, starting at 0000 universal time coordinated (UTC) and running to 2400 UTC, and likewise, extracts only the mesonet reports on the hour, thereby providing time collinear datasets. Bilinear interpolation is then applied to translate the MM5 output to mesonet sites. For vertical analysis in the MM5, the model output must first be translated from sigma levels to pressure levels in order to provide a means for direct comparison with raob data. The MM5 provides a utility called INTERPB, which performs this task. An AWK (a pattern matching language) script filters the raob files, keeping only the data from pressure levels that are available in the INTERPB output. As at the surface, bilinear interpolation of the INTERPB translated data can now be carried out at each pressure level and statistics can be generated.

The WRF output is formatted in netCDF, a more widely accepted data format. Surface statistical analysis is carried out as follows. A netCDF utility, ncdump, extracts the requested parameter values and writes them out in a FORTRAN format (i.e., in the format that a FORTRAN program would write it). A C program then extracts the first field of the ncdump output, which is a data value, and writes the data out to an ASCII file. Lastly, a FORTRAN program reads off only the hourly observations from 0000 UTC to 2400 UTC of the mesonet dataset, so as to match the hourly WRF output, and performs bilinear interpolation on the model output to collocate model data with truth data. At this point, statistical analysis may be carried out.

The WRF has no native capability, such as the MM5 INTERPB function, to transform model data to pressure levels. Thus, for MET, a group of utilities called WRFPOST, used regularly at the National Center for Environmental Prediction was adopted and successfully ported to a Linux cluster, after a number of FORTRAN coding and Makefile modifications were made. The vertical analysis for the WRF is orchestrated by a shell script and proceeds as follows: 1) the WRFPOST library of routines create a group of Gridded Binary (GRIB) files containing the WRF model data interpolated to pressure levels; 2) the Naval Research Laboratory GRIB library routine, gribsimp, is then applied to convert the GRIB files into "flat files" (i.e., each file contains the data pertaining to a particular pressure level); 3) a C routine converts the flat files into ASCII; 4) an AWK script extracts the raob data at pressure levels matching what WRFPOST extracts; and 5) the FORTRAN code performs a bilinear interpolation of the model data to raob locations at each pressure level in preparation for statistical analysis.

A number of statistical values may be derived in MET. Among them are bias (interpolated model value – truth value), root mean square error, and dispersion.

Dispersion, which is a function of the rank correlation, is given as

$$disp = \sqrt{2[1-r(f,v)]sd(f)sd(v)} \qquad (6)$$

where *f* is a single forecast variable and v is the corresponding truth data (which could also be an analysis field). Large dispersion is indicative of poor model forecast skill. The rank correlation, *r*, is given by

$$r = \sum fv / \sqrt{\sum f^2 \sum v^2} \qquad (7)$$

4. Web Services

A composition of Web services is being applied within MET. The Web provides two definitions of a Web service:

- "A Web service is a software application identified by a Uniform Resource Identifier (URI) [Request for Comment (RFC) 2396], whose interfaces and bindings are capable of being defined, described, and discovered as XML artifacts. A Web service supports direct interactions with other software agents using XML based messages exchanged via Internet-based protocols." (*7*)

- "A Web service is any piece of software that makes itself available over the Internet and uses a standardized XML messaging system. The most prominent directory of Web services is currently available via UDDI, or Universal Description, Discovery, and Integration." (*8*)

The key word in the definitions is "standardized." Any client is able to use a Web service, along with the service's methods and parameters, once the URI is known.

As described above regarding the URI, the user is queried regarding which meteorological parameter to optimize for, and the subsequent statistical analysis is then handled via Web services. Java Web services have been chosen for this application for principally two reasons: 1) if new models require testing, they can be integrated relatively simply as new Java Web services called by the Java servlet; and 2) by knowing a Web service's URI and the names of its associated methods (i.e., having a copy of the Web service's description language XML file, which fully describes the Web service), access to the Web service is made straightforward. Each of these elements is being built into a Java Web service representing one member in the phase space of possibilities. Within the Web service paradigm, a client can request information or trigger a process by calling a Web service's methods. Four Web services have been constructed for MET to carry out statistical analyses. They include a MM5-AVN Web service, a MM5-Eta Web service, a MM5-MRF Web service, and a MM5-NOGAPS Web service. Within each Web service are methods to calculate the mean bias over a 24-h forecast period for temperature; relative humidity; the *u*- and *v*-components of the wind; the wind components combined; and all parameters combined. In each case, a client will trigger a server action based on the (optimum) parameter choice.

There are many other Web services being designed and implemented. These services include a wrf-avn Web service (the WRF initialized with AVN data), a wrf-Eta Web service (the WRF initialized with Eta data), a wrf-multi-analysis-ensemble Web service (a combination of the WRF model runs using both AVN and Eta data), and a multi-model-ensemble Web service (the WRF executed with AVN or Eta data combined with the MM5 executed with AVN, Eta, MRF, or NOGAPS data). A Java Web servlet controls the execution of these Web services and will return the model producing the optimum results for the parameter of choice to the client.

Figure 2 shows the results returned when the user has selected temperature as the parameter to optimize for, and the MM5-AVN Web service has returned the minimum mean absolute error value of 0.2.


```
Model Evaluation Tool

User inputs:

   Latitude: 32.0
   Longitude: -106.0
   Number of forecast hours: 12
   Forecast hour to evaluate: 12
   Parameter to optimize for: temperature
Result
 A Model Paradigm has successfully been found. The smallest MAE for temperature was found with model paradigm MM5 AVN and the
minimum MAE is 0.2.
```

```
HTTP/1.1 200 OK
Set-Cookie: JSESSIONID=3354BEFFAB21E073621AAECD671FE941; Path=/axis
Content-Type: text/xml;charset=utf-8
Date: Mon, 01 Mar 2004 22:52:34 GMT
Server: Apache-Coyote/1.1
Connection: close

<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http
 <soapenv:Body>
  <ns1:getTemperatureMAEResponse soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:ns1="urn:mm5avn">
   <getTemperatureMAEReturn xsi:type="xsd:double">0.2</getTemperatureMAEReturn>
  </ns1:getTemperatureMAEResponse>
 </soapenv:Body>
</soapenv:Envelope>
```
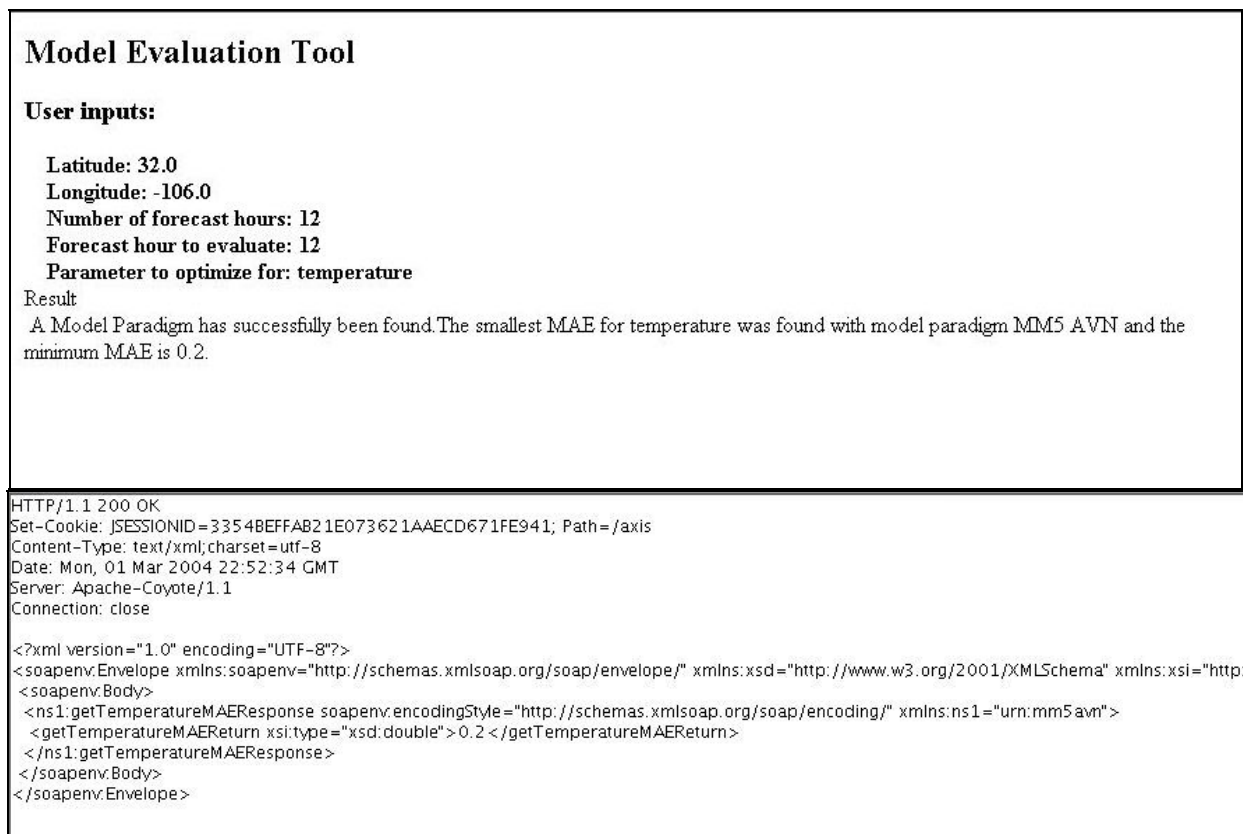
Figure 2. Results using the Web service, selecting for temperature (top window) and indicating the minimum mean absolute error for temperature from the MM5 AVN Web service while listening on port 1234 (bottom window from TCP monitor (TCPMON)).

5. Usage of XML within MET

Since XML is becoming a very widely accepted data format and the conversant language for Web services, it was decided to add to MET the tools to convert text statistical files to XML format through Java routines. After the conversion, the XML data can then be stored in a freeware XML database, called eXist. (Because future support for this freeware XML database is uncertain, this element of MET may be reconsidered.)

To then convert data extracted from the XML database and make it viewable with HTML, the XML data can be converted using JAXP tools within the JSP. XML Stylesheet Language (XSL) is applied to read and reformat the XML into HTML. A sample transformation is seen in figure 3, where a dummy dataset has been transformed from text to XML and then rendered into HTML after the application of XSL stylesheets.

Model Evaluation Tool

MODEL: *MM5-MRF*

Lead Time: 0.0

P	T rmse	T mae	T cc	RH rmse	RH mae	RH cc
1000.0	0.2	0.8	1.8	3.9	0.4	5.7
850.0	0.2	0.8	1.8	3.9	0.4	5.7
700.0	0.2	0.8	1.8	3.9	0.4	5.7
500.0	0.2	0.8	1.8	3.9	0.4	5.7
400.0	0.2	0.8	1.8	3.9	0.4	5.7
300.0	0.2	0.8	1.8	3.9	0.4	5.7
200.0	0.2	0.8	1.8	3.9	0.4	5.7
100.0	0.2	0.8	1.8	3.9	0.4	5.7
50.0	0.2	0.8	1.8	3.9	0.4	5.7
20.0	0.2	0.8	1.8	3.9	0.4	5.7
10.0	0.2	0.8	1.8	3.9	0.4	5.7

Figure 3. Sample transformation of textual XML data to HTML display.

6. XML database

As mentioned, eXist was the XML database utilized for this study. Data may be inserted into the XML database using eXist syntax; however, XPath (*2*) syntax was used to extract XML data from eXist. An example is seen below.

Here the user is requesting MM5-MRF ensemble member information for a run from May 2003 at the analysis time, and the requested statistic is bias. The `text()` portion implies returning only the text portion.

```
query.query("//model[@name='mm5-mrf']/model_run_time[@value='2003-05-
12:00:00:00']/statistic[@type='bias']/forecast[lead_time &= '0']/t/text()");
```

7. Graphical Displays

For the graphical display of the analyses, a graphics toolkit composed of Java routines, called VisAD (*1*), developed by University of Wisconsin, was employed. To render the scatterplots, a C module first reads out the "good" (non-missing) data for the parameter of interest, created either from the surface or vertical analysis. Then Java code, which uses VisAD libraries for display (to display the "truth" data versus the interpolated model data), is recompiled in order to display the correct variable with new array sizes. Finally, the Java routine is called to perform the rendering.

8. User Interface

The MET user interface is built using JSF, which is an "event-driven, component-based model for Web application development." (*9*) Tabbed panes are created with the JSF panelGrid component. A group of tabs are then set within a larger panelGrid. The panelGrid group of tabs requires a customized panel renderer (i.e., a methodology to render only the tab selected within a group). In the case of MET, a tab selection triggers a font change in the tab label and a listing of the user options for that tab.

Thus, within MET, the user is presented with tabbed panes representing user option categories under the following main headings: 1) "Model Selection/Physics" – a user selects from choices, including ensembles or single model runs, and chooses a planetary boundary layer parameterization scheme, a cumulus parameterization scheme, etc., 2) "User Settings" – the user uses this to set the model run location, number of forecast hours, and parameter to optimize for, 3) "Data Acquisition/Assimilation" – a user chooses options to select the raobs and surface data for assimilation, and 4) "Statistical Analysis/Display" – a user can select from **Scatterplots** or **Bias Plots** for display.

As an example, figure 4 represents the tabbed pane subheadings for the Model Selection/Physics heading and then shows the Planetary Boundary Layer tab as selected.

Figure 4. The menu presented to the user when the Planetary Boundary Layer tab (under the Model Selection/Physics heading) is selected.

When the user makes a selection and clicks on the **Submit Values/Check Current Settings** button, a table will display (as seen in fig. 5), indicating all of the user selections up to now. Selecting the **Home** button returns the user to the main headings pane.

User Preferences	
Model	mm5_multianalysis_ensemble
Implicit Moisture Scheme	mix_phase
Cumulus Scheme	grell
Planetary Boundary Layer	mrf
Model Run Location	
Number Forecast Hours	
Parameter to Optimize for	
Data Assimilation	
Data Acquisition	
Scatterplots	
Bias plots	

home

Figure 5. The table displayed when user chooses **Submit Values/Check Current Settings** button, indicating current user selections.

Figure 6 shows the progress bar displayed as the AVN data FTP proceeds. Note that in the upper left-hand corner of the figure are the parameters, indicating the following:

- The AVN FTP is underway.
- The Eta and MRF dataset FTP sessions are in queue.
- The current FTP session is incomplete.
- All FTP sessions are yet to be completed.

13

model_counter:0
AvnFtpStartCount:1
EtaFtpStartCount:0
MrfFtpStartCount:0
isCompletedCurrent: false
isCompletedAll: false
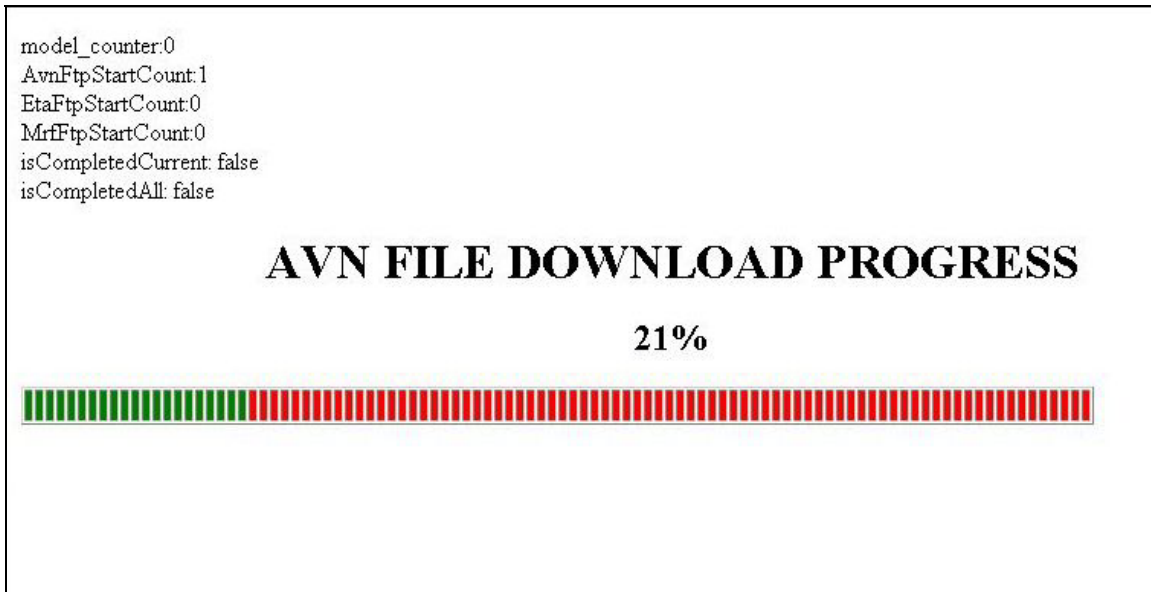
AVN FILE DOWNLOAD PROGRESS

21%

Figure 6. MET screen indicating the AVN FTP progress.

Upon successful completion of the data FTP sessions, a check is made of the status of the 32 Linux cluster nodes. This handled by the Scali tool, scaconftool. If the cluster is completely up, the user will see the display shown in figure 7.

All 32 cluster processors are up ....
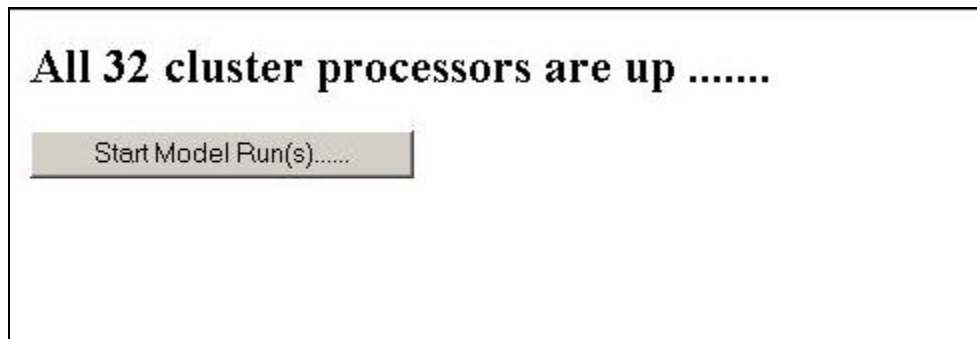
Start Model Run(s)......

Figure 7. The result of a successful system scan after all of the data FTPs are completed
and the MET has run scaconftool to determine the status of all the nodes.

If scaconftool indicates a problem with one or more cluster nodes, the user will be informed of this and advised to contact the system administrator. No other MET actions can take place until the problem is rectified. Also, as seen in figure 7, the user has the option of starting the model runs, which will yield progress bars indicating to the user the percentage of completion of the current model run.

9.   Graphical Results

Figure 8 shows a scatterplot generated by the VisAD Java routines for temperature from a WRF model run. MesoWest mesonet data from the western U.S. serves as the truth data. The y-axis

14

represents the truth values from the mesonet. The *x*-axis represents interpolated model temperature. Scatterplots can be rendered for any of the following: temperature, relative humidity, and the *u*- and *v*-components of the wind.
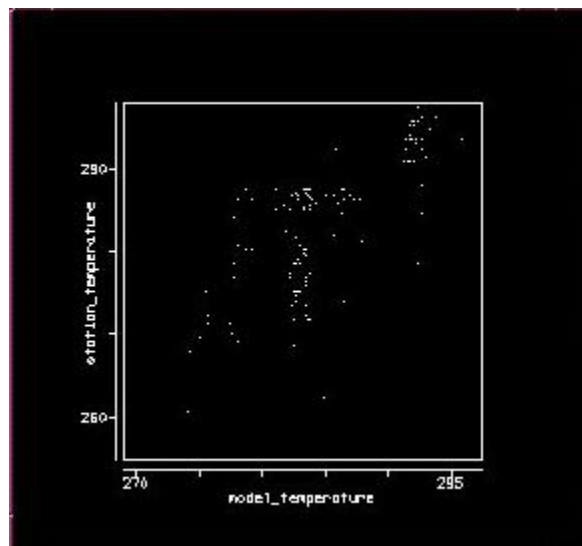

Figure 8. A scatterplot for temperature analysis at the
surface from a WRF model run.

Bias line plots covering the 24-h forecast duration may also be created for the same parameters.

10. Mobile Version of MET

Initial testing is also underway to provide mobile access to MET. With such a capability, a client could easily ascertain which model (or model ensemble) is handling recent weather conditions best and could then proceed to use that model (or model ensemble) output as input to a decision aid being used in the field. A Wireless Application Protocol (WAP) gateway is employed to allow a cell phone to communicate with the server. The WAP gateway chosen is an Open Source package created by Kannel (kannel.org) and runs on a Linux server. Figure 9 shows how a WAP stack is used as the protocol between the cell phone and the WAP gateway and contains only a few bytes per request for efficiency. The WAP gateway then translates requests to HTTP, allowing it to communicate with any HTTP server. (*10*)
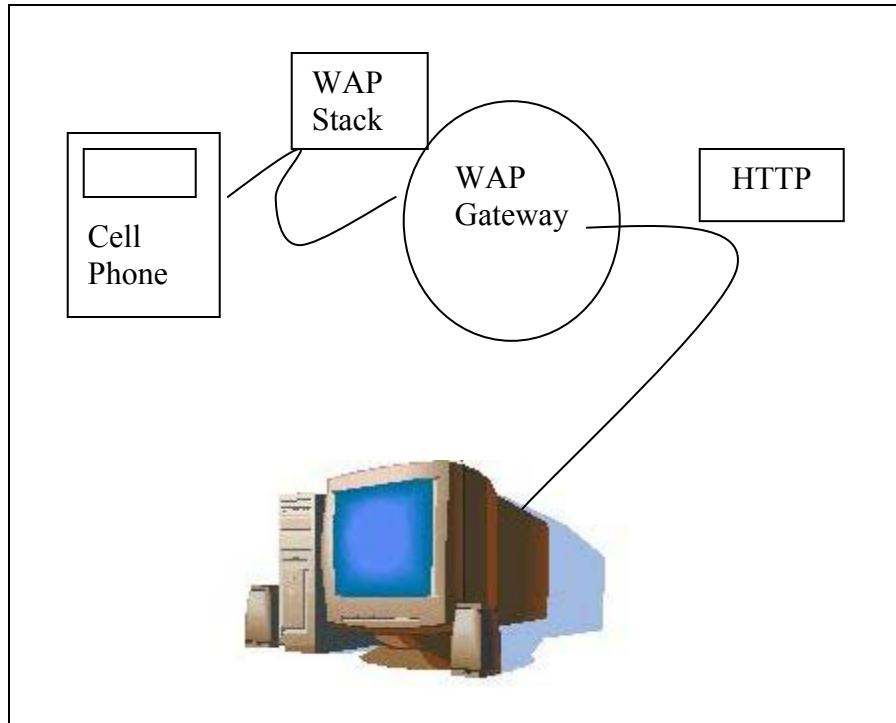
15

Figure 9. Schematic illustrating communications between mobile phone and server via WAP.

Figure 10 shows the graphic displayed on a cell phone when the user selects a model parameter to optimize for, in this case, temperature. WML cards generate the image in the cell phone display. The mobile client is able to reach the HTTP server via the Kannel WAP gateway.
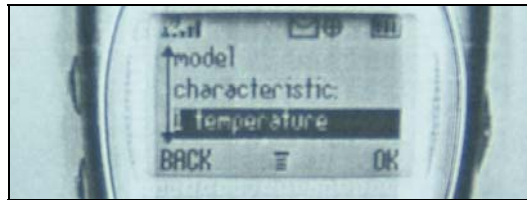

Figure 10. Screen portion of a cell phone showing
the WML card as it requests a model
parameter to optimize for from the
wireless implementation of the MET.

16

11. Conclusions

A MET has been built that allows a user to execute MM5 and WRF model runs on a Linux cluster, and analyze and display the results. JSP in conjunction with JSF, provide the control of the execution and display. These models may be run standalone or as ensembles. In addition, the user has the option of assimilating raobs and surface data.

Future enhancements to MET will include adding the ability to assimilate satellite data (for example, water vapor data from a Global Positioning System) into model runs, plus completing the "mobile MET" capability. Also, the capability for MET to gather its requisite data files from a Java Space on demand will be built.

With this initial MET prototype, the user has a platform to examine the quality of MM5 and WRF output in various model configurations. The challenge in running mesoscale models is to minimize run times, so faster platforms are always being sought out. The 32-node Linux cluster is relatively powerful; however, when all possible model configurations within MET must be run, run times become quite large. Thus, more powerful, possibly heterogeneous compute platforms will be sought out for future versions of MET.

The "mobile" version of MET provides a methodology for a user to access MET from remote locations. However, the extremely limited screen area of a cell phone is problematic, so new entities, such as Personal Digital Assistants, are being investigated for the display of MET graphical products.

References

1. VisAd Home Page. http://www.ssec.wisc.edu/~billh/visad.html (accessed June 2002–Sept 2004).

2. *Professional JSP*; 2nd Edition; WROX Press Ltd.: Birmingham, UK, 2001.

3. MesoWest mesonet Page. http://www.met.utah.edu/jhorel/html/mesonet (accessed October 2003–September 2004).

4. Granger, C.W.J. Forecasting Stock Prices: Lessons for Forecasters. *International J. of Forecasting* Royal Statistical Society (A), **1974**, *137*.

5. Gerding, S.; Myers, B. Adaptive Data Fusion of Meteorological Forecast Modules. *Proceedings from the American Meteorological Society Conference*, 2003.

6. Grimit, E. Implementation and Evaluation of a Short-Range Ensemble Forecasting System over the Pacific Northwest. Master's Thesis, University of Washington: Seattle, Washington, 2001.

7. Web Services Glossary Page (produced by Web Services Architecture Working Group). http://dev.w3.org/cvsWeb/~checkout~/2002/ws/arch/glossary/wsa-glossary.html (accessed August 2004).

8. Web Services Glossary XML Page (produced by FACTORY3X5). http://www.factory3x5.com/more_info/glossary.xml (accessed August 2004).

9. Bergsten, H. *JavaServer Faces*; O'Reilly Media, Inc.: Sebastopol, CA, 2004.

10. Kannel WAP Gateway Home Page. http://www.kannel.org (accessed October 2003–August 2004).

Acronyms

AVN	Aviation Model
API	Application Programming Interface
GRIB	Gridded Binary
HTML	Hypertext Markup Language
JAXP	Java API for XML Processing
JSF	Java Server Faces
JSP	Java Server Pages
MET	Model Evaluation Tool
METAR	Meteorological Aerodrome Report
MM5	Mesoscale Model Version 5
MRF	Medium-Range Forecasting Model
N-G	Newbold-Granger
NOGAPS	Naval Operational Global Atmospheric Prediction System
RFC	Request for Comment
URI	Uniform Resource Indicator
UTC	Universal Time Coordinated
VisAD	Visualization for Algorithm Development
WAP	Wireless Application Protocol
WML	Wireless Markup Language
WRF	Weather Research and Forecasting model
XML	Extensible Markup Language
XSL	XML Stylesheet Language

Distribution List

	Copies
US ARMY MISSILE CMND REDSTONE SCI INFO CTR AMSMI RD CS R DOC REDSTONE ARSENAL AL 35898-5241	1
PACIFIC MISSILE TEST CTR GEOPHYSICS DIV ATTN CODE 3250 POINT MUGU CA 93042-5000	1
ATMOSPHERIC PROPAGATION BRANCH SPAWARSYSCEN SAN DIEGO D858 49170 PROPAGATION PATH SAN DIEGO CA 92152-7385	1
METEOROLOGIST IN CHARGE KWAJALEIN MISSILE RANGE PO BOX 67 APO SAN FRANCISCO CA 96555	1
NCAR LIBRARY SERIALS NATL CTR FOR ATMOS RSCH PO BOX 3000 BOULDER CO 80307-3000	1
HEADQUARTERS DEPT OF ARMY DAMI POB WEATHER TEAM 1000 ARMY PENTAGON ROOM 2E383 WASHINGTON DC 20310-1067	1
HQ AFWA/DNX 106 PEACEKEEPER DR STE 2N3 OFFUTT AFB NE 68113-4039	1
AFRL/VSBL 29 RANDOLPH RD HANSCOM AFB MA 01731	1

Copies

ARL CHEMICAL BIOLOGY   1
NUC EFFECTS DIV
AMSRD ARL SL CO
APG MD 21010-5423

US ARMY RESEARCH LAB   1
AMSRD ARL SE EE
ATTN DR SZTANKAY
2800 POWDER MILL ROAD
ADELPHI MD 20783-1145

US ARMY RESEARCH LAB   1
AMSRD ARL CI
ATTN J GANTT
2800 POWDER MILL ROAD
ADELPHI MD 20783-1197

US ARMY RSRC OFC   1
ATTN AMSRD ARL RO EN
PO BOX 12211
RTP NC 27009

US ARMY CECRL   1
CRREL GP
ATTN DR DETSCH
HANOVER NH 03755-1290

ARMY DUGWAY PROVING GRD   1
STEDP MT M
ATTN MR BOWERS
DUGWAY UT 84022-5000

USAF ROME LAB TECH   1
CORRIDOR W STE 262 RL SUL
26 ELECTR PKWY BLD 106
GRIFFISS AFB ROME NY 13441-4514

US ARMY FIELD
ARTILLERY SCHOOL   1
ATSF TSM TA
FT SILL OK 73503-5600

NAVAL SURFACE WEAPONS CTR   1
CODE G63
DAHLGREN VA 22448-5000

	Copies
US ARMY OEC	1
CSTE EFS	
PARK CENTER IV	
4501 FORD AVE	
ALEXANDRIA VA 22302-1458	

US ARMY CORPS OF ENGRS   1
ENGR TOPOGRAPHICS LAB
ATTN CETEC TR G PF KRAUSE
ALEXANDRIA VA 2215-3864

US ARMY TOPO ENGR CTR   1
CETEC ZC 1
FT BELVOIR VA 22060-5546

USATRADOC   1
ATCD FA
FT MONROE VA 23651-5170

US ARMY
TRADOC ANALYSIS CMND
ATRC WSS R   1
WSMR NM 88002-5502

US ARMY RESEARCH LAB   1
AMSRD ARL CI E
COMP & INFO SCI DIR
WSMR NM 88002-55011

WSMR TECH LIBRARY BR   1
STEWS IM IT
WSMR NM 88002

US ARMY CECOM   1
INFORMATION & INTELLIGENCE
WARFARE DIRECTORATE
ATTN AMSEL RD IW IP
FORT MONMOUTH NJ 07703-5211

NAVAL RESEARCH LAB   1
MARINE METEOROLOGY DIV
7 GRACE HOPPER AVE STOP 2
MONTEREY CA 93943-5502

INTENTIONALLY LEFT BLANK.